

V77 COMPUTERS INSTRUCTION AND ASSEMBLER Reference Card

"Making machines do more so
man can do more"



SINGLE WORD ADDRESSING (FORMAT 1)

(2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Op-code				M			Address								

01-07 OXX - Direct addressing to 2048
11-17 4 - Relative to P
5 - Index by X
6 - Index by B
7 - Indirect (Max. 5 Levels)

LDA	01	ORA	11
LDB	02	ADD	12
LDX	03	ERA	13
INR	04	SUB	14
STA	05	ANA	15
STB	06	MUL	16
STX	07	DIV	17

Assembler Format

mnem addr Direct mnem addr, 2 Index by B
mnem₅ addr, 1 Index by X mnem* addr Indirect

EXTENDED ADDRESSING (FORMAT 11)

	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		6	O	Z				OP				M	
i	Address (Indirect—Max 4 Levels)														

Z = OP = 01 - 07 M = 4 - 7*
O Pre-Indexing 11 - 17
1 Post-Indexing
Op — codes and modes are the same as Single Word Addressing
*7 = Direct or Indirect

Assembler Format

mnem addr mnem* addr
mnem addr, 1 mnem* addr, 1 (Pre)
mnem addr, 2 mnem* addr, 1, 0200 (Post)

Mnemonics same as S.W.A. plus an E

REGISTER TO MEMORY V75+ (FORMAT 17)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		7						R				RX	
i	Address (Indirect—Max 4 Levels)														

LD	0070RX	AD	0072RX
ST	0071RX	SB	0073RX

Assembler Format

mnem, R addr, RX
mnem*, R addr, RX

IMMEDIATE (FORMAT 24)

(3)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		6			O	O		OP			O	O	O
	Operand														

OP = 01 - 07
11 - 17

OP Are the same op-codes as
Single Word Addressing

Assembler Format

mnem Operand

Mnemonics same as S.W.A. plus an I

IMMEDIATE V75+ (FORMAT 25)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		7			4							R	
S	Operand														

LDI	00744R
ADI	00745R

Assembler Format

LDI, R Operand
ADI, R Operand

DOUBLE PRECISION V75+ (FORMAT 20)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		4			DR							RX	
i	Address (Indirect—Max 4 Levels)														

DLD	004D0X	DSUB	004D3X
DST	004D1X	DAN	004D4X
DADD	004D2X	DOR	004D5X
		DER	004D6X

DR = 6 — DP register R0, R1
DR = 7 — DP register R4, R5

Assembler Format

mnem, 0 addr mnem*, 0 addr
mnem, 4 addr mnem*, 4 addr, RX

REGISTER CHANGE (FORMAT 6)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			5		C	T		X	B	A	X	B	A

0 0 Transfer
0 1 Increment
1 0 1's Complement
1 1 Decrement

Unconditional = 0
Execute if overflow set = 1

Bits 3, 4, 5 Source registers
Bits 0, 1, 2 Destination registers

Assembler Format

mnem
mnem Multiple source destination value

SINGLE REGISTER V75+ (FORMAT 7)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			7		4							R	

INC 00741R COM 00743R
DEC 00742R

Assembler Format

mnem, R

REGISTER TO REGISTER V75+ (FORMAT 8)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			7						RS			RD	

T 0077SD SBR 0076SD
ADR 0075SD

Assembler Format

mnem, RS, RD

BYTE V75+ (FORMAT 18)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			7		4							RX	
i	Address (Indirect—Max 1 Level)														

LBT 00746X
SBT 00747X

RX bit 0 = 0 Left Byte (15-8)
RX bit 0 = 1 Right Byte (7-0)

RX shifted right one place before Index Add

Assembler Format

mnem addr, RX
mnem* addr, RX

SHIFT (FORMAT 5)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			4		A or B	B	LEFT RIGHT	ARITH LOG	SHIFT COUNT				
							A and B	A	RIGHT	LOG					

Machine Language

LSRA 004340 ASRA 004300
LSRB 004140 ASRB 004100
LRLB 004040 ASLA 004200
LRLA 004240 ASLB 004000
LLSR 004540 LASR 004500
LLRL 004440 LASL 004400

Assembler Format

mnem Shift Count value

JUMP, JUMP-MARK, EXECUTE (FORMAT 12)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O					S	S	S	X	B	A	A	A	O
							S	S	S	=	=	=	<	>	F
							3	2	1	0	0	0	0	0	1
i	Address (Indirect—Max 4 Levels)														

OP = 1 — JUMP Bits 1 and 2
OP = 2 — JUMP AND MARK set = not condition
OP = 3 — EXECUTE

Machine Language

JMP 001000 JIF 001000
JMPM 002000 JIFM 002000
XEC 003000 XIF 003000

*Multiple conditions may be specified

Assembler Format

mnem addr JIF condition, addr
mnem* addr JIF* condition, addr

JUMP - IF V75+ (FORMAT 19)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			6		7			C				R	
i	Address (Indirect—Max 4 Levels)														

DJP 00671R (V77-800 only)
JZ 00672R JP 00675R
JNZ 00673R JDZ 00676R
JN 00674R JDNZ 00677R

Assembler Format

mnem, R addr
mnem*, R addr

JUMP AND SET RETURN (FORMAT 14)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			6		5			O				R	
i	Address (Indirect—Max 4 Levels)														

R = 5 — X register

R = 6 — B register

No indexing

Assembler Format

JSR addr, 1
JSR addr, 2
JSR* addr, 1
JSR* addr, 2

INDEXED JUMP (FORMAT 13)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			6		7			O				M	
i	Address (Indirect—Max 4 Levels)														

Post-Indexing only

M = 4 - 7*

Assembler Format

IJMP addr Modes same
IJMP addr, 1 as S.W.A.
IJMP addr, 2
IJMP* addr
IJMP* addr, 1

* 7 = Direct or Indirect

SKIP IF REGISTER EQUAL (FORMAT 16)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O			6		6			R				M	
i	Address (Indirect—Max 4 Levels)														

M = 4 - 7*

R = 1 - A register

R = 2 - B register

R = 4 - X register

Post Indexing only

Assembler Format

SRE addr, 1, 010 2 = Index by B register
SRE addr, 2, 020 * 7 = Direct / Indirect
SRE addr, 7, 040 010 = Compare A register
SRE* addr, 7, 010 020 = Compare B register
1 = Index by X register 040 = Compare X register

7

BIT TEST (FORMAT 15)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		6			4		C R	b		b b b			
i Address (Indirect—Max 4 Levels)															

If bits 5 - 4 = BT Jumps when

- 0 0 selected bit of AR = 1
 0 1 selected bit of BR = 1
 1 0 selected bit of AR = 0
 1 1 selected bit of BR = 0
 bbbb = Bit to be tested

Assembler Format

BT CR and b, addr
 BT* CR and b, addr

CONTROL (FORMAT 2 & 3)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O	O	OP CODE													
HLT	000XXX			SOF				007401							
NOP	005000			TSA				007402 (600 ONLY)							
ROF	007400														

Assembler Format

mnem
 HLT OXXX

INPUT / OUTPUT

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1		O		TYPE			func** or req						device address**		
i Address (Indirect—Max 4 levels, SEN ONLY)															

2nd word for SEN, IME or OME only

TYPE = 0 - EXC func
 1 - SEN func
 2 - INPUT REG
 3 - OUTPUT REG
 4 - EXC 2 func

*IME, OME must be a direct address

**func and device address are device independent

Assembler Format

EXC func -da
 EXC 2 func -da
 SEN func -da, addr
 IME da, addr
 OME da, addr

All others

mnem da

8

DOUBLE WORD MOVE (FORMAT 26)**V77-800 ONLY**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		6			5			M			N		
Source Address															
Destination Address															

M = Addressing Mode N = Number of Double Words Moved

- 100 Both source and destination addresses are direct.
 101 Source address is indexed by register R2 (X) and destination address is direct.
 110 Source address is direct and destination address is indexed by register R2 (X).
 111 Both source and destination addresses are indexed by register R2 (X).

Assembler Format

DMOVSD N, Saddr, Daddr
 DMOVXD N, Saddr, Daddr
 DMOV SX N, Saddr, Daddr
 DMOV XX N, Saddr, Daddr

REGISTERS LOAD, STORE (FORMAT 27)**V77-800 ONLY**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		6			5			1 or 3			RX		
i Address (Indirect—Max 4 Levels)															

1 = Registers Load
 3 = Registers Store

Assembler Format

RGLD addr RGST addr
 RGLD addr, RX RGST addr, RX
 RGLD* addr, RX RGST* addr, RX

9

BLOCK (FORMAT 2)**V77-800 ONLY**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O		O		7			4			O			X		

X = 4 block move

R0 = source starting address

R1 = destination starting address

R6 = block length

X = 6 store words

X = 7 store bytes

R1 = starting address of block

R6 = block length

Assembler Format

BMOVW
 STBYTS
 STWRDS

RTE SERVICE REQUEST MACROS**Abort Task**

label ABORT 'xx', 'yy', 'zz'

Allocate Reentrant Stack

label ALOC address

address of reentrant subroutine to
 be executed

***Allocate Memory Page(s)**

label ALOCPG n, logical addr, reject addr

Deallocate Reentrant Stack

label DEALOC

***Deallocate Memory Page(s)**

label DEALPG n, logical addr, reject addr

Delay Task

label DELAY milli, min, type

milli 5-millisecond increments

min number of minutes

type 0 = time delay expected

1 = exit & automatically rescheduled

2 = time delay or interrupt

3 = time or interrupt or I/O complete

Exit Task

label EXIT

*=VORTEX II system only

Link Background I/O

label **IOLINK** lungsd,bufloc,bufsiz
lungsd logical unit number
bufloc address of I/O buffer
bufsiz buffer size

Delay and Reload

label **LDELAY** milli,min,lun,key

*Map Memory Page(s)

label **MAPIN** n,logical addr,buffer or page,reject addr
buffer = buffer containing page numbers
page = physical page number
A = physical address
B = map buffer address

Load/Execute Overlay

label **OVLAY** type,'xx','yy','zz'
type 0 = load and execute
 1 = load and return

*Pass Map 0

label **PASS** count, from, to
count number o words
from map 0 fetch address
to user map store address

*Identify Physical Page Number

label **PAGNUM** logical addr

PIM Mask Register

label **PMSK** pim,mask,opt
pim pim number
mask interrupt lines
opt 0 = enable
 1 = disable

Resume Task

label **RESUME** 'xx','yy','zz'

Schedule Task

label **SCHED** level, wait, lun,key,'xx','yy','zz'
wait 0 = no wait, 1 = wait
'xx','yy','zz' task name

Suspend Task

label **SUSPND** susp
susp 0 = waiting for resume
 1 = waiting for interrupt
 2 = waiting for INT. or I/O complete

Set/Fetch Task TIDB Words

label **TBEVNT** value,disp,cls
value mask
disp displacement
c/s clear/set indicator

Time of Day

label **TIME**
A reg = 5-millisecond increments
B reg. = minutes

INPUT/OUTPUT CONTROL

Close File

label **CLOSE** fcb,lun,wait,mode
mode 0 = leave
 1 = update

Data Control Block

label **DCB** rl,buff,fun

@DEMAND

label **DEMAND** lun,wait

File Control Block

label **FCB** rl,buff,acc,key,'xx','yy','zz'
acc 0 = direct by logical record
 1 = sequential by logical record
 2 = direct by sector number
 3 = sequential by sector

FUNCTION

label **FUNC** dcb,lun,wait

OPEN FILE

label **OPEN** fcb,lun,wait,mode
mode 0 = rewind
 1 = not rewind

Read

label **READ** ch,lun,wait,mode
cb address of DCB or FCB
wait 1 = immediate return
 0 = suspend I/O complete
mode 0 = binary
 1 = ASCII
 2 = BCD
 3 = unformatted

@RELEASE (RELEASE)

@RESERVE (RESERV)

@RELEASE RESERVE (RELRSV)

label **COMMAND** lun,wait

@BUFFER CONTROL BLOCK

label **BCB** buf addr,buf len,offset,nxt bcb
offset displacement value of 1st word in file
 record to be read.
nxt bcb address of next BUFFER CNTL BLOCK

@ T2843/8433 ONLY

Rewind

label **REW** cb,lun,wait

Skip Record

label **SREC** cb,lun,wait,mode
mode 0 = forward
 1 = reverse

Status

label **STAT** req,err,aaa,bbb,busy
req address of I/O macro
err error
aaa EOF, BOD, BOT
bbb EOD, EOT
busy I/O not complete

Write End of File

WEOF cb,lun,wait

Write

WRITE cb,lun,wait,mode

UNIVAC V70 and 620 Series Instruction List

Instn.	Octal Code	Instn.	Octal Code	Instn.	Octal Code	Instn.	Octal Code
AD*	0072xx	IME	1020xx	JXNZM@	002046	STA	050000
ADD	120000	INA	1021xx	JXZ	001040	STAE	00605z
ADDE	00612z	INAB	1023xx	JXZM	002040	STAI	006050
ADI*	00745x	INB	1022xx	JZ*	00672x	STB	006000
ADDI	006120	INC*	0074x	SASL	004400* ⁿ	STBE	00606z
ADR*	0075xx	INR	040000	LASR	004500* ⁿ	STBI	006060
ANA	150000	INRE	00604z	LBT*	00746x	STX	070000
ANAE	00615z	INRI	006040	LD*	0070xx	STXE	00607z
ANAI	006150	IXR	005144	LDA	010000	STXI	006070
AOFA	005511	JAN	001004	LDAE	00601z	SUB	140000
AOFB	005522	JANM	002004	LDAI	006010	SUBE	00614z
AOFX	005544	JANZ@	001016	LDB	020000	SUBI	006140
ASLA	004200* ⁿ	JANZM@	002016	LDBE	00602z	T*	0077xx
ASLB	004000* ⁿ	JAP	001002	LDBI	006020	TAB	005012
ASRA	004300* ⁿ	JAPM	002002	LDI*	00744x	TAX	005014
ASRB	004100* ⁿ	JAZ	001010	LDX	030000	TBA	005021
BT@	0064vw	JA2M	002010	LDXE	00603z	TBX	005024
CIA	1025xx	JBNZ@	001026	LDXI	006030	TSAE@	007402
CIAB	1027xx	JBNZM@	002026	LLRL	004440* ⁿ	TXA	005041
CIB	1026xx	JBZ	001020	LLSR	004500* ⁿ	TXB	005042
COM*	00743x	JBZM	002020	LRLA	004240* ⁿ	TZA	005001
CPA	005211	JDNZ*	00677x	LRLB	004040* ⁿ	TZB	005002
CPB	005222	JDZ*	00676x	LSRA	004340* ⁿ	TZX	005004
CPX	005244	JMP	001000	LSRB	004140* ⁿ	XAN	003004
DADD*	004U2x	JMPM	002000	MUL	160000	XANZ@	003016
DAN*	004U4x	JN*	00674x	MULE	00616z	XAP	003002
DAR	005311	JNZ*	00673x	MULI	006160	XAZ	003010
DBR	005322	JOF	001001	NOP	005000	XBNZ@	003026
DEC*	00742x	JOFM	002001	OAR	1031xx	XBZ	003020
DER*	004U6x	JOFN@	001007	OAB	1033xx	XEC	003000
DIV	170000	JOFNM@	002007	OBR	1032xx	XOF	003001
DIVE	00617z	JP*	00675x	OME	1030xx	XOFN@	003007
DIVI	006170	JS1M	002100	ORA	110000	XS1	003100
DLD*	004U0x	JS2M	002200	ORAE	00611z	XS2	003200
DOR*	004U5x	JS3M	002400	ORAI	006110	XS3	003400
DST	004U1x	JSR@	00650x	ROF	007400	XS1N@	003106
DSUB*	004U3x	JSS1	001100	SEN	101xxx	XS2N@	003206
DXR	005344	JSS2	001200	SB*	0073xx	XS3N@	003406
ERA	130000	JSS3	001400	SBR*	0076xx	XXNZ@	003046
ERAE	00613z	JS1N@	001106	SBT*	00747x	XXZ	003040
ERAI	006130	JS2N@	001206	SOF	007401	W=0-15	Y-1.2.4
EXC	100xxx	JS3N@	001406	SOFA	005711	X=0-7	
HLT	000xxx	JS1NM@	002106	SOFB	005722	Z=4-7	
IAR	005111	JS2NM@	002206	SOFX	005744	V=0-3	
IBR	005122	JS3NM@	002406	SRE@	0066yx		
IJMP@	00670x	JXNZ@	001046	ST*	0071xx		

@ Instruction unique to 620/I, 620/I-100, and V79 Series.

* Instruction unique to V75

All MUL/DIV and Extended Addressing is optional with Univac 620/I.